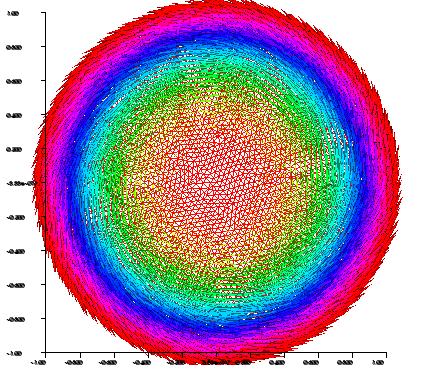
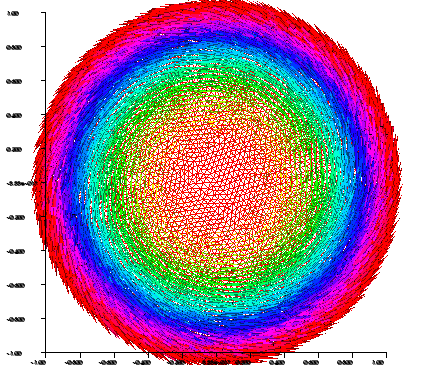
二次元A法とT法の比較



[RT0Ortho] Raviart-Thomas Orthogonal, or Nedelec finite element type I of degree 0 in dimension 2

 (3.9)

B-Field



A-Method　(P1)　　　　　　　　　　T-Method (RT0Ortho)

// Mesh

real r=1.0;

border a(t=0, 2\*pi){x=r\*cos(t); y=r\*sin(t); label=1;}

mesh disk= buildmesh(a(200));

//plot(disk);

real sigma=1.e6;

real mu=4.e-7\*pi;

real nu=1./mu;

real freq=1.0;

real s=2.\*pi\*freq;

real flux=1.0;

ofstream fid ("output.dat");

// Fespace

fespace femp1(disk, P1);

femp1 u, v;

// Problem

solve Amethod(u, v, solver="CG")

= int2d(disk)(nu\*( dx(u)\*dx(v) + dy(u)\*dy(v)) + s\*sigma\*u\*v )

+ on(1, u=s\*flux)

;

y=0.;

for(x=0.; x<=r; x +=r/100){

real by=dx(u);

cout<< "x= "<<x<< " u= "<< u << " by= " << by << endl;

fid<< "x= "<<x<< " u= "<< u << " by= " << by << endl;

}

fespace RT(disk, RT0Ortho);

RT [bx,by]=[-dy(u), dx(u)];

plot([bx,by]);

RT [hx, hy], [vx, vy];

solve Tmethod([hx, hy], [vx, vy], solver="CG")

=int2d(disk)(1./(s\*sigma)\*(dx(hy)-dy(hx))\*(dx(vy)-dy(vx)) + mu\*(hx\*vx+hy\*vy) )

+ on(1, hx=nu\*bx, hy=nu\*by)

;

[bx,by]=mu\*[hx,hy];

plot([bx,by]);

y=0;

for(x=0.; x<=r; x +=r/100){

cout<< "x= "<<x<< " by= " << mu\*hy << endl;

fid<< "x= "<<x<< " by= " << mu\*hy << endl;

}

end;